PCT/AU00/00700

10/018781

REC'D 18 JUL 2000

WIPO PCT

PRIORITY DOCUMENT

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

Patent Office
Canberra

4

I, LEANNE MYNOTT, TEAM LEADER EXAMINATION SUPPORT AND SALES hereby certify that annexed is a true copy of the Provisional specification in connection with Application No. PQ 1197 for a patent by DYNAMIC DIGITAL DEPTH RESEARCH PTY LTD filed on 25 June 1999.

COMMONWEALTH OF AUSTRALIA
AUSTRALIA
PATENT OFFICE

WITNESS my hand this
Seventh day of July 2000

LEANNE MYNOTT
<u>TEAM LEADER EXAMINATION</u>
<u>SUPPORT AND SALES</u>

AUSTRALIA

Patents Act 1990

ORIGINAL

PROVISIONAL SPECIFICATION

DDC/3 IMAGE CONVERSION AND ENCODING TECHNIQUES

The invention is described in the following statement:

DDC/3 IMAGE CONVERSION AND ENCODING TECHNIQUES

FIELD OF INVENTION

The present invention is generally directed towards stereoscopic image synthesis and more particularly toward an improved method of converting two dimensional (2D) images for further encoding, transmission and decoding for the purpose of stereoscopic display.

BACKGROUND

The applicants have previously described in PCT/AU96/00820, a method of producing left and right eye images for a stereoscopic display from an original 2D image including the steps of:

a.) identifying at least one object within the original image;

b.) outlining each object;

c.) defining a depth characteristic for each object; and

d.) respectively displacing selected areas of each object by a determined amount in a lateral direction as a function of the depth characteristic of each object, to form two stretched images for viewing by the left and right eyes in the viewer.

These steps can be individually and collectively referred to as Dynamic Depth Cuing or DDC.

Additionally, the Applicants have previously described in PCT/AU98/0155, in one aspect a method of encoding a depth map including the steps of:

a.) allocating an object number to an object;

b.) allocating the object with a depth; and

c.) defining the object outline.

The object or object outline can be defined by a series of co-ordinates, and/or curves, and in particular bezier curves where found to produce desirable results. This system could also be assisted through the use of generic libraries, both in the identification of objects within an image and also the allocation of depth of each object.

Further, the Applicants disclosed a method of transmission of the depth map information whereby the information was included in the vertical Blanking

Interval or MPEG data stream.

The Applicants' prior developments enabled an operator to simply outline an object and assign a depth to that object. This information was then processed to determine the amount of stretching required in order to place the object at the assigned depth. From an end user point of view this system is intuitive and allows the user to interactively alter the depth of an object to obtain an artistically pleasing result. From a processing point of view, the use of Bezier curves proves a highly efficient means of compressing the necessary data. However, whilst the use of Bezier curves is an efficient compressing technique, it is a relatively complex system to implement. Accordingly, a simplified technique of generating and processing depth maps is desired.

OBJECTIVE OF INVENTION

It is an objective of the present invention to provide a relatively simple technique of generating and processing depth maps, and further improve the operations of the Applicants earlier image conversion and encoding techniques.

SUMMARY OF INVENTION

With the above objectives in mind, the present invention provides in one aspect a method of encoding a depth map including:

identifying and outlining an object within an image;

allocating an object identification symbol to the object;

using the allocated object symbol to represent the shape of the object; and

allocating the object with a depth.

Advantageously, the present invention will include the further steps of:

compressing the information representing the object and its depth;

transmitting and/or storing the compressed information; and

decompressing the information.

In another aspect the present invention provides a method of encoding a depth map including:

identifying and outlining an object within an image;

allocating an object identification symbol to the object;

defining the object by drawing a plurality of lines across the image, and

determining the start and finish positions of each said line, wherein a new line is commenced each time an object boundary is reached;

allocating the object with a depth;

Preferably, the said lines will extend horizontally across the screen. However, it will be appreciated that the lines could equally extend vertically across (or up and down) the screen.

BRIEF DESCRIPTION OF THE DRAWINGS

To provide a better understanding of the present invention, reference is made to the accompanying drawings which illustrate a preferred embodiment of the present invention.

IN THE DRAWINGS:

Figure 1 shows a group of objects that have been identified, as previously disclosed and allocated a number.

Figure 2 shows how the shape of each identified object may be defined using the object number defined in Figure 1.

Figure 3 shows how the information representing this group of objects can be compressed.

Figure 4 shows an alternative method for representing each object and its depth.

DETAILED DESCRIPTION OF INVENTION

In the preferred embodiment, the image conversion technique includes the following steps:

OBJECT IDENTIFICATION

Objects in the 2D image to be converted may be identified using any of the methods previously disclosed in application PCT/AU98/01005, the contents of which are hereby included herein by reference. For illustrative purposes only, assume that the image to be converted is as per Figure 1. This image consists of four objects, namely a disk which has been allocated object number 1, a triangle that has been allocated object number 2, a square that has been allocated object number 3 and the background that has been allocated object number 4. It will be understood that the objects need not be allocated numerals, and that alphanumeric characters or any other symbol could also be used.

OBJECT REPRESENTATION

As shown in Figure 2 the shape of each of these objects within the image may be defined by using each object identification symbol, in this case a number, to "paint" the shape of the object.

As can be seen from Figure 2, object 1 has been reproduced using the symbol 1, object 2 the symbol 2, object 3 the symbol 3 and object 4 the symbol 4.

OBJECT COMPRESSION

In order to efficiently store or transmit the information representing the objects shown in Figure 1 it is necessary to compress the information. The compression and decompression process should be able to be undertaken both in software and/or hardware and operate in a fast and efficient manner.

In the case of the 2D to 3D conversion of video, where the object information is included in the Vertical Blanking Interval or MPEG stream, it is desirable that the decompression should operate sufficiently rapidly to enable real time 2D to 3D conversion of the associated 2D image.

A simple compression technique is to run length encode for each line of Figure 2. This is shown in Figure 3. The first line of Figure 2 consists of the symbol 4 repeated 50 times. Using run length encoding this would compress to

4(50)

which indicates that the symbol 4 is to be repeated 50 times along that line.

In a similar manner, each line of Figure 2 is processed to produce Figure 3.

To decompress Figure 3 the reverse process is applied.

This compression technique can effectively reduce the data required to represent Figure 1 to 20% of its uncompressed value. If desired, additional compression may be applied.

It will be appreciated by those skilled in the art that many other data compression techniques are suitable for this application. This would include, although not limited to, Lenple Ziv, Huffman and Shannon-Fano.

Once an object has been defined using the above process a differential

encoding technique can be used to eliminate the need to transfer or store data that is consistent in consecutive images. The differentially encoded data can then be compressed as previously described.

DEPTH MAPS

As previously disclosed, in order to use this representation of an object to form a depth map each object needs to be allocated a depth. The manner of defining the depth of an object is as previously disclosed in PCT/AU98/01005. Therefore, each object identified in Figure 1 will have associated with it a depth identifier in the form:

<object number> <depth flag> <depth operator>

Where <depth flag> indicates the type of depth information that follows and <depth operator> are any parameters associated with the specific type of depth flag.

This object depth information may be added to the object data stream either before or after compression.

In an alternative embodiment, the image conversion technique includes the following steps:

OBJECT IDENTIFICATION

Objects in the 2D image to be converted to 3D are identified as previously described. For purposes of explanation we will use the objects identified in Figure 1.

OBJECT REPRESENTATION

The shape of each object can be represented by drawing a series of horizontal lines across the image such that a new line is started each time an object boundary is crossed. This is illustrated in Figure 4.

The x, y co-ordinates of the starting point of the first line, marked Z1 in Figure 4 will be known and for illustrative purposes will be assigned 0,0. Likewise the end point of the line, marked Z2 in Figure 4, will be known and for illustrative purposes will be assigned 0,255.

Lines (Z1,Z2) (Z3,Z4) Z5,Z6) represent the first few lines of object 4. The values of Zn indicate the depth at which the object is to appear in the final stereoscopic image. For illustrative purposes we will assume that there are 255

possible depth values, such that depth 0 is closest to the viewer and depth 255 farthest from the viewer.

If, for example, object 4 is to appear at a constant depth 100 from the viewer then the values of Z1 and Z2 would be 100.

If, however, it was desired that object 4 should appear to ramp away from the viewer then Z1 could be set to 50 and Z2 to 100. The 2D to 3D conversion process previously disclosed in PCT/AU98/01005 would then interpolate the depth data and produce the appropriate 3D images.

Conversely, should it be desired that object 4 should appear to both ramp away from the viewer and tilt away then Zn-1 could be set to 75 and Zn set to 100.

Referring to Figure 4, the line identified as starting at Z7 would therefore consist of 3 segments as follows:

Z7 to Z8 - which defines the segment of object 4;

Z9 to Z10 - which defines the segment of object 3;

Z11 to Z12 - which defines the segment of object 4

Whilst the x, y location of points Z1 to Z7 and Z12 will be known, since they border the image, the location of points Z8 to Z11 are determined by the size of the objects being described, and can be determined from the outline of the object(s).

The allocation of depths Za and Zb will allow a linear depth ramp to be applied to an object. However, since it is desirable to add other than linear ramps to an object, for example radial ramps, the centre of the radial ramp (x, y) and its radius (z) will also need to be known.

Thus the general format for a line describing both an objects length and assigned depth is

start_depth, end_depth (length, centre (x, y), radius)

Where start_depth is relative to either the left hand edge of the image or the previous object identification and length, centre and radius may be required depending upon the length of the line (i.e. if the length is not equal to the image width) and assigned depth characteristics.

For example, using start_depths relative to the previous object, and assuming a 0 to 255 by 1 to 255 image with object 4 at a constant depth 50 and object 3 at a constant depth 20, and assuming that object 3 runs from co-

ordinate (76,4) to (200,4), the data for the first 4 lines of Figure 4 will be:

Line 1	50
Line 2	50
Line 3	50
Line 4	50, 50, 75 20, 20 125 50, 50, 55

This data can be compressed as previously described or using other compression techniques familiar to those skilled in the art. Similarly, the allocation of a depth for each object is as previously disclosed.

ALTERNATIVE PROCESS FOR DEPTH ASSIGNMENT

An alternative method of assigning the depth of an object is to "paint" the object using a graphics paint brush or air brush, such as used in the Adobe Photoshop graphics software, with a colour equal to the depth that is to be assigned to the object.

For illustrative purposes assume that there are 256 different depth levels available with 0 representing an object closest to the observer and 255 furthest away from the observer. Assume that depth 0 is assigned to white and 255 to black then intermediate depths will be assigned a shade of grey.

Considering Figure 1, assume that object 1 is to be set to depth 10. The operator selects depth 10 from a pallet which determines a corresponding shade of grey to assign to the graphics paint brush or air brush.

The operator then paints the area of object 1 with the brush, thus allocating depth 10 to the object. Since the operator can see the results of his actions on the computer screen it is possible to accurately select objects.

Different size brushes, or air brush patterns, may be selected to allow fine detail or rapid fill of large areas. Errors can be corrected using a graphical eraser such as found, for example, in Adobe Photoshop.

Once the shape of an object has been painted using the brush the depth of the object can be altered if necessary by altering the shade of grey. Standard techniques such as found in graphics drawing packages can be used to change the shade of grey.

Using a mouse, or other locating device, it is possible to add variations in object depth within an object. The variations in depth are applied by selectively changing the shade of grey within the object. Examples of the depth variations

include, but are not limited to, linear ramps, non-linear ramps and radial ramps.

Painting the object with a shade of grey has been chosen for illustrative purposes only. In practice any colour, shape or symbol may be used to enable the operator to paint the selected object.

A preferred embodiment includes separating the steps of painting the object and selecting the depth of the object. An effective process is to cause the object being painted to take on transparent or glass like texture. This enables the operator to see areas of the object that have already been painted rather than obscuring them with a solid colour. Once the object has been painted in this manner a solid depth colour can be applied.

ALTERNATIVE PROCESS FOR CREATING BEZIER CURVES

Since the peripheral edges of the object are known it is possible to locate the boundary of the object and automatically construct Bezier curves that trace the peripheral of the object. The Bezier curves may be adjusted as necessary to cause exact alignment of the object and also adjusted to allow for movement of the object as it moves in successive frames.

ALTERNATIVE PROCESS FOR TRANSPORT OF DDC DATA

In previous applications, it has been assumed that the Dynamic Depth Cueing (DDC) data has been imbedded in the original 2D image in either the Vertical Blanking Interval or MPEG stream.

In practice this is a convenient way to transport and store the DDC encoded 2D images. However, it will be understood by to those skilled in the art that the DDC data and original 2D images can be stored and transported separately if desired.

For example, if the original 2D image where a video sequence that was held on a data server accessible via the Internet then the DDC data could be accessed from the same server or an entirely different server in a geographically different location. The user could then download the video sequence and DDC data relating to the video sequence independently and combine them either at the time of viewing or prior to viewing.

The application of the DDC data to the original video sequence to create 3D can be undertaken in real time, as the viewer looks at the images, or off-line

prior to viewing. The creation of 3D from the DDC data and the original 2D image can be undertaken in either software or hardware as previously disclosed.

In relation to software conversion from 2D to 3D, a convenient way to achieve this on a Personal Computer (PC) is to provide a software plug-in for an existing 2D viewer. For example, a plug in for the Apple QuickTime movie viewer would be suitable for general PC viewing as would plug-ins for Internet Web browser such as Internet Explorer and Netscape Communicator. These plug-ins could be most conveniently provided to the viewer by downloading over the Internet.

Alternatively a custom software application could be developed and conveniently provided to the user via a download from the Internet or other efficient method.

USE OF DDC FOR THE 2D TO 3D CONVERSION OF STILL IMAGES

A subset of the use of DDC data for the conversion of 2D moving images into 3D is the conversion of still images. The original image may be in any digital format and may have originated from either a film camera, with the resulting image being scanned to produce a digital file, or directly from a digital still or moving camera.

For transfer between Personal Computers and in particular transfer over the Internet, it is desirable that the DDC data is imbedded in a standard image format. Such formats are know to those skilled in the art and include, but are not limited to, JPEG, BMP, GIFF and TIFF.

JPEG and GIFF are compressed formats that also enable the insertion of private data within the image files. In a preferred embodiment the DDC data is included in the private data area of these compressed image formats.

This process therefore enables an image containing DDC data to be viewed in either 2D, if the user does not have either a suitable viewer or plug-in, or 3D if a viewer or plug-in is available.

As previously disclosed, DDC data may be used to create both a stereo pair of images and also a number of stereo pairs suitable for use with autostereoscopic displays using lenticular lenses. An example of such a display

is that manufactured by Philips which requires 7 or 9 pairs of stereo images.

For use with a still image, the DDC data can be used to produce multiple monoscopic or stereoscopic views from an original 2D image. In a preferred embodiment such images could be viewed on a Personal Computer using a Windows interface with the viewpoint selected by moving a horizontal slider bar.

DATED this 25th day of June, 1999.
DYNAMIC DIGITAL DEPTH RESEARCH PTY LTD

WATERMARK PATENT & TRADEMARK ATTORNEYS
4TH FLOOR, "DURACK CENTRE"
263 ADELAIDE TERRACE
PERTH W.A. 6000
AUSTRALIA

Figure 1

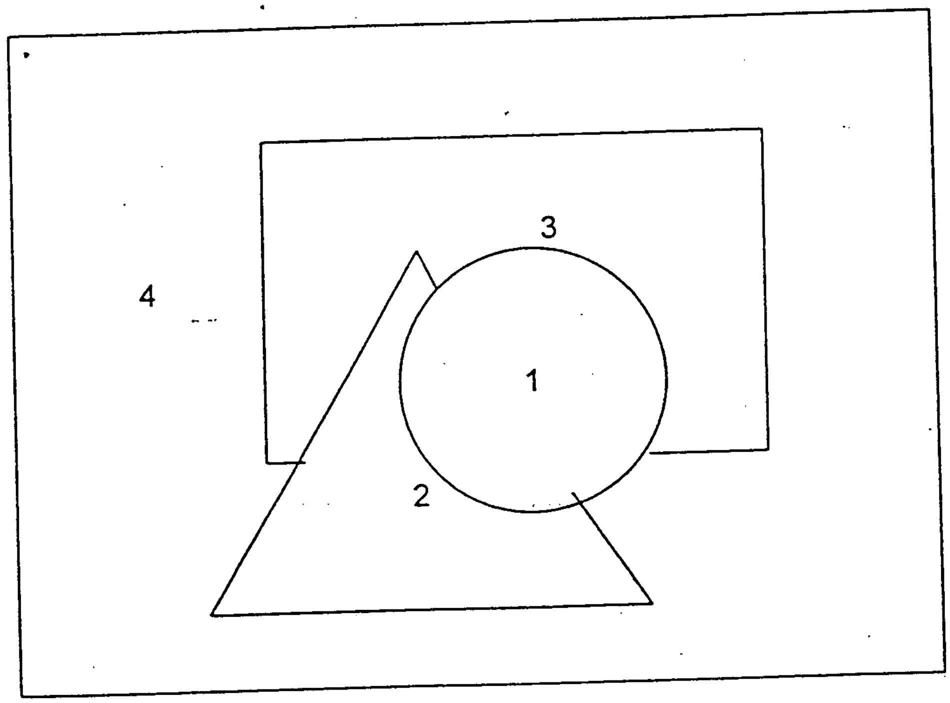

Figure 2

```
44444444444444444444444444444444444444444444444444
44444444444444444444444444444444444444444444444444
44444444444444444444444444444444444444444444444444
44444444444444444444444444444444444444444444444444
44444444444444444444444444444444444444444444444444
4444444444·3333333333333333333333333331444444444444
4444444444·33333333333 '33333333333333331444444444444
4444444444·333333333 2 '3 1111111  333331444444444444
4444444444·33333333 22  1111111111  3331444444444444
4444444444·3333333'222 11111111111111 331444444444444
4444444444·3333332222.1111111·11111111,31444444444444
4444444444·33333 22222 11111111111111 331444444444444
4444444444·333 22222222. 1111111111 .3331444444444444
4444444444·33'2222222222.1111111 .333331444444444444
44444444444/2222222222222222222444444444444444444
4444444444 22222222222222222222224444444444444444444
444444444 22222222222222222222222 144444444444444444
44444444444444444444444444444444444444444444444444
44444444444444444444444444444444444444444444444444
```

Figure 3

4(50)					
4(50)					
4(50)					
4(50)					
4(50)					
4(10)	3(28)	4(12)			
4(10)	3(28)	4(12)	1(7)	3(5)	4(12)
4(10)	3(10)	2(1)	1(11)	3(3)	4(12)
4(10)	3(9)	2(2)	1(14)	2(3)	4(12)
4(10)	3(7)	2(3)	1(16)	3(1)	4(12)
4(10)	3(6)	2(4)	1(14)	3(2)	4(12)
4(10)	3(5)	2(6)	1(11)	3(3)	4(12)
4(10)	3(3)	2(8)	1(7)	3(3)	4(12)
4(10)	3(2)	2(11)			
4(11)	2(20)	4(19)			
4(10)	2(22)	4(18)			
4(9)	2(24)	4(17)			
4(50)					
4(50)					

Figure 4